The Minmax Multidimensional Knapsack Problem with Application to a Chance-Constrained Problem

Moshe Kress,[1] Michal Penn,[2] Maria Polukarov[2]

[1] *Operations Research Department, Naval Postgraduate School, Monterey, California*

[2] *Faculty of Industrial Engineering and Management, Technion, Haifa, Israel*

Received 13 December 2004; revised 25 March 2007; accepted 31 March 2007
DOI 10.1002/nav.20237
Published online 17 May 2007 in Wiley InterScience (www.interscience.wiley.com).

Abstract: In this paper we present a new combinatorial problem, called minmax multidimensional knapsack problem (MKP), motivated by a military logistics problem. The logistics problem is a two-period, two-level, chance-constrained problem with recourse. We show that the MKP is NP-hard and develop a practically efficient combinatorial algorithm for solving it. We also show that under some reasonable assumptions regarding the operational setting of the logistics problem, the chance-constrained optimization problem is decomposable into a series of MKPs that are solved separately. © 2007 Wiley Periodicals, Inc. Naval Research Logistics 54: 656–666, 2007

Keywords: knapsack problem; chance constraints; recourse; military logistics; combinatorial algorithm

1. INTRODUCTION

We present a new combinatorial model called minmax multidimensional knapsack problem (MKP). The MKP is motivated by a military logistics optimization problem typical to tactical-level ground operations such as supply and resupply of ammunition to an artillery battalion in a 2-day operation. The problem is a two-period stochastic programming problem with recourse. However, under our reasonable operational assumptions, it is shown to have unique features that reduce it into two separate and sequential (albeit not independent) sets of MKPs. We show that the MKP is NP-hard and develop a new practically efficient algorithm for solving it.

Stated simply, the military logistics problem is to find minimum cost inventories that satisfy minimum responsiveness requirements. The requirements are expressed in terms of probabilities for satisfying demands in a two-period operation of a military unit that comprises several weapons. While some inventories must be determined before the operation, others can be set after the demands in the first period are realized—a situation that lends itself to a two-period, chance-constrained, stochastic programming model with recourse.

Stochastic programming models with recourse have been applied in supply-chain and related problems, mostly in the context of optimizing expected values. Escudero et al. [9] utilize scenario modeling for production and capacity planning. Their multiperiod model minimizes expected costs, while considering several recourse alternatives. Dempster et al. [7] consider a multiperiod supply-chain scheduling problem where demands and costs are uncertain. In a recent paper, Cattani et al. [4] analyze the simultaneous production of two products: a market-specific product tailored to the needs of individual regions, and a global product that could be sold in many regions. They consider a two-stage model with additional, post-recourse, uncertainty in the second stage, and seek to maximize an expected profit function. Cheung and Powell [6] consider the class of multistage dynamic networks with random arc capacities, and propose a successive convex approximation approach for the expected recourse function, which captures the future effects of current decisions under uncertainty. This method decomposes the network in each stage into sub-problems for which expected recourse functions are easy to obtain [5]. Recently, Gupta et al. [11] used chance constraints programming approach coupled with two-stage stochastic programming with recourse methodology to construct a two-stage supply chain plan under demand uncertainty with continuous random variables. The authors utilized linear programming duality to obtain the expectation of the recourse function associated with the second supply chain stage in terms of the first stage production decisions.

One-period chance-constrained (OPCC) problems of the form $\min\{cx : P[Tx \geq q] \geq p,\ Ax \geq b,\ x \geq 0\}$ are

Correspondence to: M. Kress (mkress@nps.navy.mil)

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
25 MAR 2007	**N/A**	**-**

4. TITLE AND SUBTITLE
The Minmax Multidimensional Knapsack Problem with Application to a Chance-Constrained Problem

5a. CONTRACT NUMBER
5b. GRANT NUMBER
5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

5d. PROJECT NUMBER
5e. TASK NUMBER
5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School Department of Operations Research Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
In this paper we present a new combinatorial problem, called minmax multidimensional knapsack problem (MKP), motivated by a military logistics problem. The logistics problem is a two-period, two-level, chance-constrained problem with recourse. We show that the MKP is NP-hard and develop a practically efficient combinatorial algorithm for solving it. We also show that under some reasonable assumptions regarding the operational setting of the logistics problem, the chance-constrained optimization problem is decomposable into a series of MKPs that are solved separately.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**SAR**	**11**	

well studied for continuous random variables q. However, there are only a few papers investigating the discrete random variables case, e.g. [3, 8, 16]. In [16] the author considers an integer version of the OPCC problem (IP-OPCC) with more general objective functions, and uses methods of disjunctive programming to approximate the convex hull of the feasible region. For some special cases a full description of the convex hull is given. In [8] the concept of *p-efficient points* (PEP) is introduced and used for deriving lower and upper bounds for the OPCC problem with discrete probability distributions. Let F denote the joint distribution function of q, then for $0 \leq p \leq 1$, the PEP are defined as the minimal realization q points (in the sense of the order $\leq$) for which $F(q) \geq p$. In [2] a branch and bound algorithm that utilizes PEP is presented. In [15] a generalization of the IP-OPCC is analyzed where new valid inequalities related to the precedence constrained knapsack polyhedra are developed and are used by a general iterative algorithm to solve the generalized problem. In these methods an optimal solution can be obtained by exploring the set of PEP. However, since the set of PEP can be extremely large, the optimization procedure may encounter computational difficulties. Nevertheless, "good" solutions can be derived by looking at a subset of PEP.

In this paper we use a different IP formulation to model the OPCC problem with discrete probability distributions, and observe that the resulting IP problem is a special case of a combinatorial problem termed *minmax multidimensional knapsack problem* (MKP). Given a set $A = \{a^s : s \in S\}$ of n-dimensional vectors a^s of real numbers, each associated with a weight p_s, the MKP consists of selecting a subset $A' \subseteq A$ such that $\sum_{a^s \in A'} p_s$, the weight of A', exceeds a certain threshold, and the objective is to minimize the sum of the componentwise maxima of the vectors in A'.

The one-dimensional MKP is related to the *min knapsack problem* [1], where one looks for a subset of items such that the sum of their weights exceeds a given constant, and the sum of their values is minimized. While both problems have the same cover type constraint, they differ in the objective function; in the MKP we look for a minmax solution while in the min knapsack problem we seek a minimum sum solution. The different objective functions affect the complexity of the problems; while the min knapsack problem is known to be NP-hard (see for example [1]) the one-dimensional MKP can be solved by a simple greedy polynomial time algorithm. Another remotely related problem is the NP-hard *multidimensional knapsack problem* which also consists of selecting A', a subset of a given set of items, such that the total value of A' is maximized while a set of knapsack constraints are to be satisfied (e.g. [14] and [3]). We note that the multidimensional knapsack problem differs from the MKP in both the constraints and the objective function. The constraint in the MKP is of a covering type while the constraints of the multidimensional knapsack problem are of a packing type. Also the objective function of the multidimensional knapsack problem is to maximize a total value while in the MKP it is a minmax objective function. To the best of our knowledge this is the first time the MKP is defined and solved. We believe that the MKP and its algorithm are interesting by themselves and moreover, are useful for solving a class of chance-constrained problems.

The rest of the paper is organized as follows. In Section 2 we motivate the modeling effort and introduce some notation. In Section 3 we introduce the single-period chance-constrained optimization problem, and in Section 4 we describe the MKP and develop an algorithm for solving it. We show that the single-period chance-constrained problem is an MKP. In Section 5 we formulate the two-period chance-constrained optimization problem and show that this formulation is equivalent to an IP model, which is decomposed into a series of MKPs. Summary and conclusions are given in Section 6. The Appendix contains proofs for the decomposition property.

2. MOTIVATION AND NOTATION

Our model is motivated by a military logistics problem typical to ground operations. Consider a battalion that comprises several weapons (e.g., artillery pieces). Each weapon consumes ammunition from a designated attached stockpile. There is also a considerable amount of ammunition on-board the weapon itself [13]; however this inventory is an emergency safety stock, to be used only if the stockpile is empty. The total amount of ammunition in the stockpile and on board the weapon is assumed to be enough to satisfy any foreseeable demand but the objective is to tap the on-board inventory as little as possible and to rely only on the stockpile [10]. An additional ammunition depot is attached to the battalion headquarters at the rear of the combat zone.

We consider a 2-day operation. During the first day a weapon can use ammunition only from its attached stockpile or, if needed, from its own safety stock. It cannot rely on logistical support from other sources because movement in the combat zone during the combat operation is risky. At the end of the first day, after demands have been observed, depleted stockpiles and possibly reduced safety stocks are replenished from the depot or from other weapons' stockpiles that transship surplus ammunition. The replenishment process is completed before the second day of operation. The replenished amount of each weapon must cover possible expenditure from the weapon's safety stock during the first day and the (yet unknown) demand in the second day. Similar inventory control situations may occur in the retail industry too.

The weapons are called henceforth *demand points* (DP). We require that the supplies allocated to the DPs on each one of the two days satisfy the demand during that day with a given minimum probability. Because of the inherent scarcity of relevant demand data regarding military operations [12], demand distributions are generated based on expert inputs in the form of combat scenarios. In the one-period problem, let $D = (D_1, D_2, \dots, D_n)$ denote the vector of random variables that represents the demands at DP_i, $i = 1, \dots, n$, and let $d = (d_1, \dots, d_n)$ denote a realization of this vector, which is called a *demand scenario*. We assume a finite number of demand scenarios where S denotes their index set. Let $Pr[D = d^s] = p_s > 0, s \in S$, where $\sum_{s \in S} p_s = 1$. In the two-period problem we have two vectors of random variables, and accordingly, two index sets of demand scenarios, S^1 and S^2.

Let x_i denote the amount of supply initially allocated to DP_i, and let Y denote the amount of supply initially deployed in the depot. The parameters C_{x_i} and C_Y are the cost of a unit of supply at DP_i and the depot, respectively. These costs are incurred at the first period. There are no additional costs in the problem. Military DPs are similar units (e.g., artillery guns) and therefore it is reasonable to assume that $C_{x_i} = C_X$ for all $i = 1, \dots, n$.

The decisions made at the beginning of day 1 are with respect to (a) the amount of supply x_i (stockpile) to be allocated to each DP, and (b) the total amount of supply Y to be kept in the depot at the beginning of period 1, to be used, if necessary, at the second period. The recourse variables are the shipments that take place at the beginning of day 2 from and to the depot. For a given pair of distributions of interrelated discrete demand random variables—one for each day—the objective is to find a minimum cost inventory policy such that demands are satisfied in both days with probabilities that exceed certain thresholds. From now on we use the terms "day" and "period" interchangeably. We begin with the single-period problem.

3. THE SINGLE PERIOD PROBLEM

The problem is to minimize the total amount of supply such that a certain level of *logistics responsiveness* is attained. The logistic responsiveness is measured by the probability that *all* demands are satisfied, that is, none of the weapons has to use ammunition from its safety stock. Formally,

$$\min \sum_{i=1}^{n} x_i \tag{1}$$

$$s.t.$$

$$Pr[D \le x] \ge Q, \tag{2}$$

$$x \ge 0, \ \ x \in \mathbb{Z}^n. \tag{3}$$

where $x = (x_1, \dots, x_n)$.

Constraint (2) is the logistics responsiveness chance constraint and Q is the probability threshold set by the commander. Since we assume that the demand scenarios are discrete random variables, problems (1)–(3) may be formulated as the following IP problem:

$$\min \sum_{i=1}^{n} x_i \tag{4}$$

$$s.t.$$

$$x_i - d_i^s \delta_s \ge 0, \quad i = 1, \dots, n, \ s \in S, \tag{5}$$

$$\sum_{s \in S} p_s \delta_s \ge Q, \tag{6}$$

$$x_i \ge 0, \ \ x_i \in \mathbb{Z}, \ \ \delta_s \in \{0, 1\}, \quad i = 1, \dots, n, \ s \in S. \tag{7}$$

The binary variables δ_s, $s \in S$, indicate whether a certain scenario has been selected to be satisfied. Constraint (6) guarantees that the probability of unsatisfied demand does not exceed the operationally set threshold $1 - Q$.

Problems (4)–(7) is an MKP, a newly defined combinatorial problem which is formulated next.

4. THE MKP

Let $a^s = (a_1^s, \dots, a_n^s)$ be an n-dimensional real vector, and let $A = \{a^s : s \in S\}$ be a set of $|S|$ such vectors. Each vector a^s in A has weight (value) $p_s > 0$. Let $A' \subseteq A$ be a subset of A. Define

$$G(A') = \sum_{i=1}^{n} \max_{a^s \in A'} a_i^s,$$

where $\max_{a^s \in A'} a_i^s$ is the largest ith component among the vectors in A'.

A subset $A' \subseteq A$ is said to be *Q-feasible* for a given parameter Q if $\sum_{a^s \in A'} p_s \ge Q$. The *minmax multidimensional knapsack problem* (MKP) is to find a Q-feasible subset $A^* \subseteq A$ such that $G(A^*)$ is minimal. The subset A^* is called an *optimal minmax subset*.

EXAMPLE 1: Let $A = \{(1, 5), (5, 3), (3, 7), (10, 1), (7, 7)\}$, $Q = 0.8$ and $p_s = 0.2$ for all s. Then any subset of A containing at least 4 elements is feasible, and an optimal minmax subset of A is $A^* = \{(1, 5), (5, 3), (3, 7), (7, 7)\}$, with $G(A^*) = 7 + 7 = 14$.

It is easily seen that problem (4)–(7) is actually an MKP, where a^s is a demand scenario d^s, $s \in S$, and the objective is to find a Q-feasible subset of these scenarios such that the sum of the componentwise maxima of the corresponding demand vectors is minimized.

Next we develop a simple combinatorial algorithm, termed the MKP-Algorithm, for solving the MKP. The case $n = 1$ is trivial. A greedy algorithm solves the problem by simply arranging the elements in A in a nondecreasing order $s(1), \ldots, s(|S|)$. The optimal subset is $A^* = \{a^{s(1)}, a^{s(2)}, \ldots, a^{s(j^*)}\}$, where j^* is the smallest index such that $\sum_{j=1}^{j^*} p_{s(j)} \geq Q$.

The case $n > 1$ is more complex. The algorithm comprises two stages: a preliminary stage and a main stage. In the preliminary stage we discard from A all vectors that are evidently nonoptimal. In the main stage the remaining subset is "pruned" and an optimal minmax subset is obtained. That is, the algorithm finds and then deletes, a complement of a Q-feasible subset, to obtain a required optimal minmax subset.

4.1. Preliminary Stage

Let $a^s_{\text{sum}} = \sum_{i=1}^n a^s_i$, $s \in S$, and, without loss of generality, assume that $a^1_{\text{sum}} \leq a^2_{\text{sum}} \leq \cdots \leq a^{|S|}_{\text{sum}}$. Let j^* be the smallest index such that $\sum_{j=1}^{j^*} p_s \geq Q$. Define the set B as follows:

$$B = \left\{ a^s \in A : \ a^s_{\text{sum}} \leq \sum_{i=1}^n \max\left\{a^1_i, \ldots, a^{j^*}_i\right\} = G(\{a^1, \ldots, a^{j^*}\}) \right\}.$$

Note that $\{a^1, \ldots, a^{j^*}\}$ is a subset of B. For an empty set we let $j^* = 0$.

PROPOSITION 4.1: If $a^s \in A \setminus B$ for some $s \in S$ then $a^s \notin A^*$, where A^* is any optimal minmax subset.

PROOF: Suppose $a^{s'} \in A \setminus B$ and $a^{s'} \in A^*$ for some optimal minmax subset A^*. Then,

$$G(A^*) = \sum_{i=1}^n \max_{a^s \in A^*} a^s_i \geq a^{s'}_{\text{sum}}.$$

Since $a^{s'} \in A \setminus B$, it follows that

$$a^{s'}_{\text{sum}} > \sum_{i=1}^n \max\left\{a^1_i, \ldots, a^{j^*}_i\right\} = G(\{a^1, \ldots, a^{j^*}\}),$$

where $\{a^1, \ldots, a^{j^*}\}$ is a feasible subset. Thus, A^* cannot be an optimal minimax subset, in contradiction. □

We conclude that an optimal minmax subset of A must be a subset of B.

4.2. Main Stage

We search for an optimal minmax subset by considering its possible complement subsets in B. In order to reduce the number of subsets examined, we generate Ψ, a family of subsets, in which a complement of an optimal minmax subset is guaranteed to be included. We describe the construction of Ψ more precisely below.

For each i, $i = 1, \ldots, n$, we arrange the vectors of B in a nonincreasing sequence B_i according to their ith component. That is,

$$a^{s_i(1)} \succ a^{s_i(2)} \succ \cdots \succ a^{s_i(|B|)} \quad \text{if and only if } a^{s_i(1)}_i \geq a^{s_i(2)}_i \geq \cdots \geq a^{s_i(|B|)}_i,$$

and B_i is the sequence $(a^{s_i(1)}, a^{s_i(2)}, \ldots, a^{s_i(|B|)})$.

Then, we define Ψ to be the family of all possible subsets ψ_k, $k = 1, \ldots, |\Psi|$, of B that satisfy the following two conditions:

(1) ψ_k is of the form

$$\psi_k = \bigcup_{i \in \{1,\ldots,n\}} \left\{a^{s_i(1)}, \ldots, a^{s_i(m_i)}\right\},$$

where $0 \leq m_i \leq |B|$. Thus, each ψ_k is a prefix, that is, a union of truncated sequences from B_i, $i = 1, \ldots, n$. Note that $m_i = 0$ means that no prefix is taken from the ith order.

(2) ψ_k is a maximal cardinality set that satisfies

$$p(A \setminus B) + \sum_{a^s \in \psi_k} p_s \leq 1 - Q,$$

where $p(A \setminus B) = \sum_{a^s \in (A \setminus B)} p_s$. That is, the complement in A of the union of the set of deleted vectors (which are evidently nonoptimal) and ψ_k, is a Q-feasible subset of A. Adding one more vector from B to ψ_k will violate the inequality and therefore will render the complement infeasible.

Now, for each $\psi_k \in \Psi$ define

$$M(\psi_k) = \sum_{i=1}^n \max_{a^s \in (B \setminus \psi_k)} a^s_i = G(B \setminus \psi_k)$$

and let

$$\hat{k} \in \arg\min\{M(\psi_k) : \ \psi_k \in \Psi\}.$$

The set $B \setminus \psi_{\hat{k}}$ is an optimal minmax subset of A. Before we prove the validity of the MKP-Algorithm, consider Example 2.

Table 1. Scenarios.

s	a_1^s	a_2^s
1	3	5
2	6	1
3	8	9
4	6	5
5	3	5
6	4	9
7	2	7
8	7	2
9	10	10
10	4	3

EXAMPLE 2: Let $S = \{1, \ldots, 10\}$, $n = 2$, $Q = 0.6$, and $p_s = 0.1$ for all $s \in S$ (see Table 1).

First, we rank the vectors in a nondecreasing order of the sum of their components (see Table 2).

$$
\begin{aligned}
j^* = 6; s(1) = 2, s(2) = 10, s(3) = 1,&\\
s(4) = 5, s(5) = 7, s(6) = 8;&\\
\max\left\{a_1^2, a_1^{10}, a_1^1, a_1^5, a_1^7, a_1^8\right\}&\\
+ \max\left\{a_2^2, a_2^{10}, a_2^1, a_2^5, a_2^7, a_2^8\right\} &= 7 + 7 = 14.
\end{aligned}
$$

Proposition 4.1 shows that a^3 and a^9 can be deleted from further consideration since the sums of their components are 17 and 20, respectively, which are greater than 14. Thus, $B = \{a^1, a^2, a^4, a^5, a^6, a^7, a^8, a^{10}\}$.

Next we generate two rankings of B, according to each component of a^s. Each ranking is in a nonincreasing order of the corresponding component (see Table 3).

Now, $\Psi = \{\{(7,2), (6,1)\}, \{(7,2), (4,9)\}, \{(4,9), (2,7)\}\}$, and $M(\{(7,2), (6,1)\}) = 6 + 9 = 15$, $M(\{(7,2), (4,9)\}) = 6 + 7 = 13$ and $M(\{(4,9), (2,7)\}) = 7 + 5 = 12$.

We conclude that the optimal minmax subset A^* comprises the vectors with the indices $1, 2, 4, 5, 8, 10$, and $G(A^*) = 12$. Note that the optimal minmax subset in this example is unique. This is not necessarily true in general.

Recall from Proposition 4.1 that any optimal minmax subset of A is contained in B. The next proposition shows the existence of an optimal subset such that its complement in B lies in Ψ.

PROPOSITION 4.2: There exists an optimal minmax subset $A^* \subseteq B$ such that $B \setminus A^* \in \Psi$.

PROOF: Let $W \subseteq B$ be the complement of an optimal minmax subset in B, and suppose that $W \notin \Psi$. Define,

$$
E(W) = \left\{a^s \in W : \forall i \text{ there exists } a^{s_i} \in (B \setminus W) \text{ and } a_i^{s_i} \geq a_i^s\right\}.
$$

That is, if $a^s \in E(W)$, then for each B_i, at least one of the components of the truncated sequence $(a^{s_i(1)}, a^{s_i(2)}, \ldots, a^s)$ is not contained in W. Let $\mathcal{W}$ be the set of all such W and let $\hat{W}$ be such that

$$
|E(\hat{W})| = \min\{|E(W)| : W \in \mathcal{W}\}.
$$

If $|E(\hat{W})| = 0$ we are done. Otherwise, there is a vector $a^{\hat{s}} \in E(\hat{W})$. According to the definition of $E(W)$, for each $i = 1, \ldots, n$ there exists $a^{s_i} \in B \setminus \hat{W}$ such that $a_i^{s_i} \geq a_i^{\hat{s}}$. But then,

$$
\sum_{i=1}^{n} \max\left\{a_i^s : a^s \in B \setminus \hat{W}\right\} = \sum_{i=1}^{n} \max\left(\left\{a_i^s : a^s \in B \setminus \hat{W}\right\} \cup \left\{a_i^{\hat{s}}\right\}\right),
$$

which implies that $E(\hat{W})$ is not minimal, in contradiction. Therefore, there exists an optimal minmax subset whose complement belongs to Ψ. □

Thus, the algorithm shown above solves the MKP. Recall that as defined in [8], for any distribution function F of q, and any $0 \leq p \leq 1$, the p-efficient points (PEP), are defined as the minimal realization q points (in the sense of the order $\leq$) for which $F(q) \geq p$. Notice that any PEP is p-feasible but not vice versa. The MKP can be solved by the more general and powerful methods which are based on polyhedral techniques as described previously in the Introduction and presented in [2, 8, 15, 16]. However, our MKP-Algorithm, which examines points in B—possibly some non-PEP—is conceptually simpler and easier to use than those methods.

4.3. Complexity and Computational Analysis

THEOREM 4.3: MKP is NP-complete and the complexity of the MKP-Algorithm is $O(n^k)$, where

$$
k = \min\left\{\left\lceil \frac{1 - Q}{\min_s p_s} \right\rceil, |S|\right\}.
$$

Table 2. Rank order of the sums.

s	a_1^s	a_2^s	a_{sum}^s
2	6	1	7
10	4	3	7
1	3	5	8
5	3	5	8
7	2	7	9
8	7	2	9
4	6	5	11
6	4	9	13
3	8	9	17
9	10	10	20

PROOF: We prove the NP-completeness by showing a simple reduction to the knapsack problem. Recall that the knapsack problem is defined as follows. Given non-negative numbers c_i, w_i, $i = 1, \ldots, n$, and b, find a subset $T \subseteq \{1, \ldots, n\}$ such that $\sum_{i \in T} w_i \le b$ and $\sum_{i \in T} c_i$ is maximized. We show that the knapsack problem polynomially transforms to MKP using the following reduction. Let $|S| = n$ and let A be the set of n vectors of dimension n, where the ith vector is set to be $c_i \times e_i$ with e_i the ith unit vector. Let $p_i = w_i$ and $Q = \sum_{i=1}^{n} w_i - b$. Then, if $A^* \subseteq A$ is an optimal Q-feasible subset then $A \setminus A^*$ is an optimal solution for the knapsack problem, and vice versa. We turn now to show the complexity of the MKP-Algorithm.

Summing up the n components for each one of the vectors in A takes $O(n|S|)$ computations, arranging the sums in a nonincreasing sequences takes $O(|S| \log |S|)$ steps and constructing the set B takes $O(|S|)$ steps. Thus, the *preliminary stage* of the algorithm takes $O(n|S|) + O(|S| \log |S|)$ steps. The *main stage* of the algorithm requires arranging n sequences, an operation that requires $O(n|S| \log |S|)$ computations. Let k be an upper bound on the size of the largest set in Ψ. Observe that k is the minimum between the number $|S|$ of possible vectors in A, and the maximal number of vectors that can "squeeze" in any ψ_k. This maximal number is determined by the smallest value among $\{p_s : s \in S\}$ and the threshold Q, and is given by $\lceil \frac{1-Q}{\min_s p_s} \rceil$. Finally, examining all possible subsets in Ψ takes $O(n^k)$. This can be done by considering vectors of length of at most k, in which the jth component is a number $1 \le i \le n$ indicating according to which prefix the vector (scenario) was chosen to be included in Ψ_k. Also, after each step, the set of possible vectors (scenarios) in ψ_k is updated by removing the already chosen vectors (scenarios). Thus, since $O(n^k) \ge O(n|S| \log |S|)$, the complexity of the MKP-Algorithm is $O(n|S|) + O(|S| \log |S|) + O(n|S| \log |S|) + O(n^k) = O(n^k)$. □

Note that examining all possible subsets of Ψ can also be done by checking all prefixes of length $0 - k$ for each i. This takes a time of at most $O(k^n)$. However, since in practical military (and many commercial) logistics problems k is a relatively small integer, we have chosen to present the complexity in terms of $O(n^k)$ rather than $O(k^n)$. The probability threshold Q is usually larger than 0.9, and relevant scenarios have typically probability that is at least 0.01. Therefore, in such situations $k \le 10$, which implies the high efficiency of our algorithm for any practical purposes. Note, as well, that for a fixed k or a fixed n, our MKP-Algorithm runs in polynomial time.

Table 3. Rank order according to the first component.

s	a_1^s	a_2^s
8	7	2
2	6	1
4	6	5
6	4	9
10	4	3
1	3	5
5	3	5
7	2	7

Table 4. Average running times (in seconds) of the MKP-Algorithm and the IP code.

	MKP		IP	
$\lvert S\rvert$	Average	S.D.	Average	S.D.
40	0.07	0.06	0.62	0.12
60	1.44	2.66	3.88	2.09
100	71.41	76.79	375.67	392.10

Next, the MKP-Algorithm is compared to a general IP code (CPLEX 8.0) with respect to running time. The comparison has been executed on Intel Pentium 4, 2Ghz CPU, 512Mb RAM, run under Windows XP. The MKP-Algorithm has been implemented in Microsoft Visual C++ 6.0.

We fix the number of DPs at $n = 15$, which is approximately the number of battalions in a division, the number of batteries in an artillery regiment, and the number of artillery pieces in a battalion. The number of scenarios ranges between $|S| = 40$ and $|S| = 100$. The scenarios are uniformly distributed (all scenarios are equally probable) and the probability threshold is 0.9. For each value of $|S|$, 100 randomly generated problems were solved by the MKP-Algorithm and the general IP code. Table 4 presents the average running times (in seconds) and the corresponding standard deviations.

For this range of data the MKP-Algorithm clearly outperforms the general IP code. This result is reversed if n gets larger. For $|S| = n = 50$, the general IP code is 10 times faster than the MKP-Algorithm. However, in practical military logistics problems, in which n is relatively small (e.g. an artillery battalion comprises 16 pieces), the MKP-Algorithm is more efficient.

5. THE TWO-PERIOD MODEL

Recall that x_i is the amount of supply initially allocated to DP_i, $i = 1, \ldots, n$, and Y is the amount of supply deployed initially in the depot. These are the first-stage decision variables. The second stage (recourse) variables are shipments between the depot and the DPs that take place after the first day demands are realized. Suppose that scenario $s \in S^1$ has been realized in the first day. For that particular scenario s, and the set of demand scenarios for day 2, with their corresponding probabilities, let y_i^s, $i = 1, \ldots, n$, denote a feasible flow of supply between the depot and DP_i such that the demand in day 2 is satisfied with probability not smaller than a given threshold. If $y_i^s \geq 0$, then the depot sends out supply to DP_i, and if $y_i^s \leq 0$, the depot receives surplus supply back from DP_i. The sum $y^s = \sum_{i=1}^n y_i^s$ denotes the total net flow of supply between the depot and the DPs at the end of the first day, given scenario s has been realized. This amount is to be shipped out before the beginning of day 2, to satisfy the demand in the second day. If $y^s < 0$, then there is a back flow of supply from the DPs to the depot at the end of the first day, which means that the supplies left in the DPs after the first day are more than enough for satisfying the demand in day 2. Let $y = \max_{s \in S^1}\{y^s\}$. If y is positive, then it is the amount of supply that is stored in the depot at the beginning of day 1, that is,$Y = \max\{0, y\}$.

Consider the following example. Suppose $n = 2$ and there are three equally likely demand scenarios for day 1: $(100, 0)$, $(0, 150)$, and $(30, 40)$ for $s = 1, 2, 3$, respectively. These three demand scenarios must be satisfied with certainty ($Q_1 = 1$). Suppose that for day 2 there are two equally likely scenarios: (80,0) and (0,70) that must be satisfied with certainty. Clearly, $x_1 = 100$ and $x_2 = 150$ is a feasible (and minimum) solution for the first day. Suppose scenario 1 has occurred, then $y_1^1 = 80$ and $y_2^1 = -80$ is a feasible flow of supply between the depot and the two DPs at the end of day 1. In fact, these flows are also minimal in the sense that the depot must ship at least 80 units to DP_1, and it cannot receive more than 80 units from DP_2. Here $y^1 = 0$. If scenario 2 has occurred then $y_1^2 = -20$ and $y_2^2 = 70$ with $y^2 = 50$ are minimum feasible flows, which imply that the depot must have at least 50 units in order to satisfy the required demand in day 2. If scenario 3 has occurred, then $y_1^3 = 10$ and $y_2^3 = -40$ with $y^3 = -30$. In this case ($s = 3$) the net flow is negative and therefore there is no need for supplies stored in the depot. In this example, $y = \max\{0, 50, -30\} = 50$ and $Y = \max\{0, 50\} = 50$.

For simplicity of exposition, as long as it does not cause any confusion, we have chosen to indicate by the index s the variables and the data that correspond to the first day and by the index t those that correspond to the second day. Accordingly, p_{st} is the conditional probability of scenario $t \in S^2$ given that scenario $s \in S^1$ has been realized in day 1. The two-period problem is described as follows: In period 1, before observing the demands in day 1, we determine the inventories x_i, $i = 1, \ldots, n$, in the DPs and the inventory Y in the depot. At the end of day 1, after observing the demand scenario s that has been realized in day 1, we determine the shipments y_i^s to and from the depot to be carried out before day 2. Excess supply at a certain DP is shipped back to the depot, and shortage in supply at a ceratin DP is replenished from the depot. The net flow of these shipments determine the inventory Y in the depot, which operates also as a clearing house for the DPs. Recall that there are no shipment costs. The initial allocation to the DPs x_i, $i = 1, \ldots, n$, must satisfy the demands in day 1 with probability not smaller that Q_1 – see constraints (9)–(11) below. For <u>each</u> realized scenario in day 1, including a scenario s' for which $d_i^{s'} > x_i$ for some i (in which case the safety stock in DP_i is used), if such exists, we require that (a) all on-board safety stocks are replenished to their full initial capacity, and (b) the demands in day 2 are satisfied with probability not smaller than Q_2. These requirements are manifested in constraints (12)–(14) below. The two-stage optimization model is formulated as follows:

$$\min \; C_X X + C_Y Y \tag{8}$$

$$s.t.$$

$$x_i - d_i^s \delta_s \geq 0, \quad i = 1, \ldots, n, \; s \in S^1, \tag{9}$$

$$X - \sum_{i=1}^n x_i \geq 0, \tag{10}$$

$$\sum_{s \in S^1} p_s \delta_s \geq Q_1, \tag{11}$$

$$y_i^s - \left(d_i^s + d_i^t - x_i\right) + M(1 - \delta_{st}) \geq 0, \quad i = 1, \ldots, n, \; s \in S^1, \; t \in S^2, \tag{12}$$

$$Y - \sum_{i=1}^n y_i^s \geq 0, \quad s \in S^1, \tag{13}$$

$$\sum_{t \in S^2} p_{st} \delta_{st} \geq Q_2, \quad s \in S^1, \tag{14}$$

$$x_i, Y \geq 0, \;\; x_i, y_i^s \in \mathbb{Z}, \;\; \delta_s, \delta_{st} \in \{0, 1\}, \quad i = 1, \ldots, n, \; s \in S^1, \; t \in S^2. \tag{15}$$

The binary variables δ_s, $s \in S^1$, and δ_{st}, $s \in S^1, t \in S^2$ indicate whether a certain scenario has been selected to be satisfied in day 1, and in day 2 given a scenario in day 1, respectively. Recall that the requirements in day 2 must be satisfied with respect to each possible realization of the demand scenario in day 1. The variables y_i^s, $i = 1, \ldots, n$, are the recourse variables: the flow of supply to (or from) DP_i, given that scenario s has been realized in day 1. Recall that the value of some y_i^s may be negative, in which case supply is actually taken away from DP_i. The constraints in (12) determine the demand scenarios in day 2 to be satisfied, where M is a large constant compared to the demand data. That is, if $\delta_{st} = 1$ then the demand of scenario t is to be satisfied for each DP_i, given scenario s occurred on the first day. Since inventories cannot be negative, we require that $x_i \geq 0$, $i = 1, \ldots, n$, $Y \geq 0$. The following example demonstrates this model.

Table 5. Scenarios in day 1.

Scenario	DP_1	DP_2	DP_3	Pr.
1	10	20	30	0.3
2	30	60	20	0.4
3	20	50	10	0.3

EXAMPLE: There are $n = 3$ DPs, and on each day of the two-days operation there are three possible demand scenarios, as shown in Tables 5–7.

The probabilities of the scenarios in day 1 are given in Table 5, and the conditional transition probabilities from scenarios in day 1 to scenarios in day 2 are presented in Table 7. The demand scenarios of day 1 and day 2 are given in Tables 5 and 6, respectively. The costs of a unit supply are $C_X = 5$ and $C_Y = 6$ for a DP and the depot, respectively. The required minimum probability thresholds are $Q_1 = 1$ and $Q_2 = 0.7$, for day 1 and day 2, respectively. An optimal deployment (there may be multiple optima) is $x_1 = 50$, $x_2 = 80$, $x_3 = 60$ with $X = \sum_{i=1}^{3} x_i = 190$, and $Y = 0$. If scenario 1 is realized in day 1, then all three scenarios can be satisfied in day 2 (with probability 1); if scenario 2 is realized in day 1, then scenarios 2 and 3 are satisfied in day 2 (with probability $0.4 + 0.3 = 0.7$ as required); if scenario 3 is realized in day 1, then once again all three scenarios can be satisfied in day 2 (with probability 1). The values of the recourse variables if scenario 1 is realized are: $y_1^1 = -10$, $y_2^1 = -50$ and $y_3^1 = 20$, which means that some of the excess supplies in DP_i, $i = 1, 2$ is distributed to DP_3. Similarly, $y_1^2 = 0$, $y_2^2 = -10$, $y_3^2 = 10$, and $y_1^3 = 0$, $y_2^3 = -20$, $y_3^3 = 0$. Notice that the depot is initially empty ($Y = 0$). This comes with no surprise because it is cheaper to store supplies in the DPs than in the depot ($C_X < C_Y$). Absent transportation costs in the second period, it would always be better to allocate all the needed supplies to the DPs. If the costs are reversed, that is, $C_X = 6$ and $C_Y = 5$, then an optimal solution is $x_1 = 30$, $x_2 = 60$, $x_3 = 30$ with $X = \sum_{i=1}^{3} x_i = 120$, and $Y = 70$. It is shown below that the problem can be decomposed into a series of MKPs and that the optimal deployment is independent of the actual values of the costs and depends only on the ordinal relation between C_X and C_Y.

Next we show that the two-period problems (8)–(15) can be decomposed into two separate problems, each is an MKP.

Table 6. Scenarios in day 2.

Scenario	DP_1	DP_2	DP_3
1	30	10	10
2	20	10	50
3	10	10	20

Table 7. Conditional transition probabilities.

Scenarios in day 1 Scenarios in day 2	1	2	3
1	0.7	0.3	0.2
2	0.2	0.4	0.7
3	0.1	0.3	0.1

The period 1 problem, which is labeled Φ_1 and is equivalent to (4)–(7), is

$$
\begin{gathered}
\min\ X \\
s.t. \\
x_i - d_i^s \delta_s \geq 0, \quad i = 1, \dots, n,\ s \in S^1, \\
X - \sum_{i=1}^{n} x_i \geq 0, \\
\sum_{s \in S^1} p_s \delta_s \geq Q_1, \\
x_i \geq 0, x_i \in \mathbb{Z}, \delta_s \in \{0, 1\}, \quad i = 1, \dots, n, s \in S^1.
\end{gathered}
$$

Note that since $d_i^s \geq 0$ for all i and $s \in S^1$, the non-negativity requirements on the x_i variables are redundant. For any given feasible solution $x = (x_1, \dots, x_n)$ of Φ_1, the period 2 problem, $\Phi_2(x)$, is,

$$
\begin{gathered}
\min\ y(x) \\
s.t. \\
y_i^s(x) + M(1 - \delta_{st}) \geq \left(d_i^s + d_i^t - x_i\right), \\
i = 1, \dots, n,\ s \in S^1,\ t \in S^2, \\
y(x) - \sum_{i=1}^{n} y_i^s(x) \geq 0, \quad s \in S^1, \\
\sum_{t \in S^2} p_{st} \delta_{st} \geq Q_2, \quad s \in S^1, \\
y_i^s(x) \in \mathbb{Z},\ \delta_{st} \in \{0, 1\}, \quad i = 1, \dots, n,\ s \in S^1,\ t \in S^2.
\end{gathered}
$$

Note that $y(x)$ represents the total sum of shipments to and from the depot and can be negative.

Theorem 5.1 below shows that an optimal solution of the two-period problems (8)–(15) is obtained by solving Φ_1 and $\Phi_2(x)$ sequentially. Recall that the optimal deployment is independent of the actual values of C_X and C_Y. Moreover, the optimal total amount of supply $(X+Y)$ is constant for a particular problem instance and is independent of the costs C_X and C_Y. This property suggests the model is robust in the sense that it exempts military planners from specifying exact cost values which are difficult to estimate. They need to express just ordinal preferences between two logistic options. Also, Proposition 5.2 below indicates that the integrality constraints on the x and the y variables in these problems are redundant if the demand values d are integer. We leave the proofs of these results to the Appendix.

THEOREM 5.1: Let $\hat{x}$ be an optimal solution for Φ_1 and $\hat{y}(\hat{x})$ be the corresponding optimal objective value of $\Phi_2(\hat{x})$, and let $\hat{X} = \sum_{i=1}^{n} \hat{x}_i$ and $\hat{Y}(\hat{x}) = \max\{0, \hat{y}(\hat{x})\}$. Then,

(i) if $C_X \geq C_Y$ then $(\hat{X}, \hat{Y}(\hat{x}))$ is optimal for (8)–(15);
(ii) if $C_X < C_Y$ then $(\hat{X} + \hat{Y}(\hat{x}), 0)$ is optimal for (8)–(15).

For each of the problems Φ_1 and $\Phi_2(x)$ denote by MIP-Φ_1 and MIP-$\Phi_2(x)$, respectively, the former problems where the integrality requirements are not imposed on the x and the y variables.

PROPOSITION 5.2: Let d^s and d^t be integral demand vectors for each $s \in S^1$ and $t \in S^2$, respectively. Then, there always exist optimal solutions of MIP-Φ_1 and of MIP-$\Phi_2(x)$ which are integral.

We have already argued in Section 3 that Φ_1 is essentially an MKP. We wish to show now that a solution for $\Phi_2(x)$ is obtained by solving $|S^1|$ MKPs. For a given period 1 solution x and a given period 1 scenario $s \in S^1$, consider problem $\Phi_2^s(x)$ below:

$$\min \sum_{i=1}^{n} y_i^s(x)$$
$$s.t.$$
$$y_i^s(x) + M(1-\delta_{st}) \geq \left(d_i^s + d_i^t - x_i\right), \quad i = 1, \ldots, n, t \in S^2,$$
$$\sum_{t \in S^2} p_{st}\delta_{st} \geq Q_2,$$
$$y_i^s(x) \in \mathbb{Z}, \ \delta_{st} \in \{0,1\}, \quad i = 1, \ldots, n, \ t \in S^2.$$

Clearly, each one of the $|S^1|$ problems $\Phi_2^s(x)$ is an MKP.

PROPOSITION 5.3: For a given period 1 solution x, let $\hat{y}^s(x) = (\hat{y}_1^s(x), \ldots, \hat{y}_n^s(x))$ be an optimal solution of $\Phi_2^s(x)$, $s \in S^1$, and let $\hat{s} \in \arg\max\{\sum_{i=1}^{n} \hat{y}_i^s(x) : s \in S^1\}$. Then, $\hat{y}(x)$, an optimal solution of $\Phi_2(x)$ is obtained by,

$$\hat{y}(x) = \sum_{i=1}^{n} \hat{y}_i^{\hat{s}}(x).$$

PROOF: The objective in problem $\Phi_2(x)$ is to select the minimum $y(x)$ value that satisfies the probability threshold requirement (14) for all period 1 scenarios $s \in S^1$. This minimum is attained if for each problem $\Phi_2^s(x)$, $s \in S^1$, $\sum_{i=1}^{n} y_i^s(x)$ is minimized. $\hat{y}(x)$ is the maximum among these values. □

6. SUMMARY AND CONCLUSIONS

In this paper we present a new knapsack-related combinatorial problem termed minmax multidimensional knapsack problem (MKP) that is motivated by a military logistics problem. The logistics problem is to determine an optimal deployment of inventories that satisfies certain operational requirements in a two-days scenario. Recourse opportunities, which are crucial in combat related military environments, are explicitly incorporated in our model. We show that the resulting two-period stochastic-programming problem can be solved by solving a series of MKPs. A practically efficient algorithm is developed for solving the MKP.

The above result can be generalized to the multi-period problem with recourse. The multi-period problem can be decomposed into T separate problems, $\Phi_1, \Phi_2, \ldots, \Phi_T$, that correspond to periods $1, 2, \ldots, T$, respectively. We observe that to solve the multi-period problem we need to optimize only the first and the last periods' allocations. For all intermediate periods we need to find just feasible solutions that satisfy the corresponding responsiveness chance constraints. This observation follows from the replenishment policy, where all tapped safety stocks are replenished in the subsequent periods, and the assumption regarding free transshipment.

Possible extensions of the logistics model involve additional constraints such as zero safety stock, which excludes back-orders, and no transshipment, which implies that the recourse variables must be non-negative. Another possible extension may be to incorporate possible (stochastic) interdiction on the transportation of supplies from depot and among the DPs. These extensions may generate extensions to the MKP where side constraints are incorporated in the knapsack-like setting.

APPENDIX

We prove now Theorem 5.1 and Proposition 5.2. First we show that the resulting sequential optimal solutions for Φ_1 and Φ_2 provide an optimal solution for the combined problem. To prove this result we need the following lemma.

LEMMA A1: Let x' and x'' be two feasible solutions for Φ_1 such that $x_i'' = x_i' + \alpha_i$, $\alpha_i \in \mathbb{R}$, $i = 1, \ldots, n$. Let $\hat{y}(x')$ and $\hat{y}(x'')$ be the corresponding optimal objective values of $\Phi_2(x')$ and $\Phi_2(x'')$, respectively. Then, $\hat{y}(x'') = \hat{y}(x') - \sum_{i=1}^{n} \alpha_i$.

PROOF: We start by showing that $\hat{y}(x'') \geq \hat{y}(x') - \sum_{i=1}^{n} \alpha_i$. Let x be any feasible solution of Φ_1 and let $a_i^{s,t}(x) = d_i^s + d_i^t - x_i$ for any $i = 1, \ldots, n$,

$s \in S^1$, $t \in S^2$. Then, for each $s \in S^1$ we define

$$A^s(x) = \{a^{s,t}(x) = (a_i^{s,t}(x))_{i=1}^n : t \in S^2\}.$$

Let $\hat{a}^s(x)$ be an optimal solution to the MKP on $A^s(x)$ and observe that $\hat{y}(x) = \max\{\hat{a}^s(x) : s \in S^1\}$. Then it suffices to show that for each $s \in S^1$, the inequality $\hat{a}^s(x'') \geq \hat{a}^s(x') - \sum_{i=1}^n \alpha_i$ holds. This is because $\hat{y}(x'') \geq \hat{a}^s(x'') \geq \hat{a}^s(x') - \sum_{i=1}^n \alpha_i$ for each $s \in S^1$ and $\hat{y}(x') = \max\{\hat{a}^s(x') : s \in S^1\}$. Let s be any scenario in S^1. Assume on the contrary that $\hat{a}^s(x'') < \hat{a}^s(x') - \sum_{i=1}^n \alpha_i$, then $\sum_{i=1}^n (\hat{a}_i^s(x'') + \alpha_i) < \sum_{i=1}^n \hat{a}_i^s(x')$. We show below that $\hat{a}^s(x'') + \alpha$, where $\alpha = (\alpha_i)_{i=1}^n$, is a feasible solution for the MKP on $A^s(x')$, contradicting the optimality of $\hat{a}^s(x')$.

Let $\hat{a}^s(x'')$ be an optimal solution, and hence a feasible solution, for the MKP on $A^s(x'')$. Therefore, there exists a feasible subset $W \subseteq S^2$ such that for each $t \in W$ and for each i, the inequality $\hat{a}_i^s(x'') \geq a_i^{s,t}(x'')$ holds. Also, for each $t \in W$ and for each i, by the definition of $a_i^{s,t}(x)$ and of x'' the equality $a_i^{s,t}(x'') = a_i^{s,t}(x') - \alpha_i$ holds. Hence, for each $t \in W$ and each i, $\hat{a}_i^s(x'') \geq a_i^{s,t}(x'') = a_i^{s,t}(x') - \alpha_i$, implying $\hat{a}_i^s(x'') + \alpha_i \geq a_i^{s,t}(x')$. Thus, $\hat{a}^s(x'') + \alpha$ is a feasible solution for the MKP on the set $A^s(x')$, of value less than $\hat{a}^s(x')$, contradicting the optimality of $\hat{a}^s(x')$.

Showing the other direction that $\hat{y}(x'') \leq \hat{y}(x') - \sum_{i=1}^n \alpha_i$ can be done in the same way by showing that $\hat{y}(x') \geq \hat{y}(x'') - \sum_{i=1}^n (-\alpha_i)$. Thus, by the two directions we get that $\hat{y}(x'') = \hat{y}(x') - \sum_{i=1}^n \alpha_i$ as required. □

The next theorem states that the sequential procedure for obtaining the logistics deployments in periods 1 and 2 is globally optimal.

THEOREM 5.1: Let $\hat{x}$ be an optimal solution for Φ_1 and $\hat{y}(\hat{x})$ be the corresponding optimal objective value of $\Phi_2(\hat{x})$, and let $\hat{X} = \sum_{i=1}^n \hat{x}_i$ and $\hat{Y}(\hat{x}) = \max\{0, \hat{y}(\hat{x})\}$. Then,

(i) if $C_X \geq C_Y$ then $(\hat{X}, \hat{Y}(\hat{x}))$ is optimal for (8)–(15);
(ii) if $C_X < C_Y$ then $(\hat{X} + \hat{Y}(\hat{x}), 0)$ is optimal for (8)–(15).

PROOF: (i) To show the correctness of the theorem in the case of $C_X \geq C_Y$ it suffices to show that for any feasible solution x of Φ_1 such that $X = \hat{X} + \sum_{i=1}^n \alpha_i$ with $\sum_{i=1}^n \alpha_i \geq 0$, where $X = \sum_{i=1}^n x_i$, the inequality $C_X X + C_Y \hat{Y}(x) \geq C_X \hat{X} + C_Y \hat{Y}(\hat{x})$ holds.

Suppose $\hat{Y}(x) = 0$, which yields $\hat{y}(x) \leq 0$. If $\hat{Y}(\hat{x}) = 0$ then the above inequality follows from the fact that $X \geq \hat{X}$. Otherwise, $\hat{Y}(\hat{x}) > 0$, implies $\hat{y}(\hat{x}) > 0$. By Lemma A1, $\hat{y}(\hat{x}) - \sum_{i=1}^n \alpha_i = \hat{y}(x) \leq 0$, which implies $\sum_{i=1}^n \alpha_i \geq \hat{y}(\hat{x}) = \hat{Y}(\hat{x})$. Then,

$$\begin{aligned} C_X X + C_Y \hat{Y}(x) &= C_X \hat{X} + C_X \sum_{i=1}^n \alpha_i + C_Y \cdot 0 \\ &\geq C_X \hat{X} + C_X \hat{Y}(\hat{x}) \geq C_X \hat{X} + C_Y \hat{Y}(\hat{x}). \end{aligned}$$

Suppose now $\hat{Y}(x) > 0$, which implies that $\hat{Y}(x) = \hat{y}(x) > 0$. By Lemma A1, since $X \geq \hat{X}$, we get $\hat{Y}(x) = \hat{Y}(\hat{x}) - \sum_{i=1}^n \alpha_i$. Then,

$$\begin{aligned} C_X X + C_Y \hat{Y}(x) &= C_X \hat{X} + C_X \sum_{i=1}^n \alpha_i + C_Y \hat{Y}(\hat{x}) - C_Y \sum_{i=1}^n \alpha_i \\ &= C_X \hat{X} + C_Y \hat{Y}(\hat{x}) + (C_X - C_Y) \sum_{i=1}^n \alpha_i \geq C_X \hat{X} + C_Y \hat{Y}(\hat{x}). \end{aligned}$$

(ii) In the case of $C_X < C_Y$ we only need to show that $C_X X + C_Y \hat{Y}(x) \geq C_X(X + \hat{Y}(x))$ for any feasible solution x of Φ_1. The above follows directly from $C_X \leq C_Y$. □

We next turn to show that the integrality requirements on the x and the y^s variables in our IP formulations are redundant. For each of the problems Φ_1 and $\Phi_2(x)$ denote by MIP-Φ_1 and MIP-$\Phi_2(x)$, respectively, the former problems where the integrality requirements are not imposed on the x and the y^s variables.

PROPOSITION 5.2: Let d^s and d^t be integral demand vectors for each $s \in S^1$ and $t \in S^2$. Then, there always exist optimal solutions of MIP-Φ_1 and of MIP-$\Phi_2(x)$ which are integral.

PROOF: The proposition follows immediately from the fact that the δ_s, δ_t variables are $\{0, 1\}$-variables and the integrality of the demand vectors. Observe that in problem MIP-Φ_1 we seek for a subset of scenarios (this is since δ_s is a binary variable) for which the required integral demands will be satisfied. The above coupled with the minimality of X imply the integrality of the x's. Similarly, it can be shown that MIP-$\Phi_2(x)$ is integral for any integral vector x and integral demand vector. □

ACKNOWLEDGMENTS

We are grateful to Tal Raviv for conducting our computational experiments and for his valuable suggestions that significantly improved the presentation of our paper. We also thank Ittai Avital for his insightful comments.

REFERENCES

[1] R. Bar-Yehuda, Using homogeneous weights for approximating the partial cover problem, J Algorithms 39 (2000), 137–144.
[2] P. Beraldi and A. Ruszczyński, A branch and bound method for stochastic integer problems under probabilistic constraints, Optim Methods Software 17 (2002), 359–382.
[3] D. Bertsimas and R. Demir, An approximate dynamic programming approach to multi-dimensional knapsack problem, Manage Sci 48 (2002), 550–565.
[4] K. Cattani, G. Ferrer, and W. Gilland, Simultaneous production of market-specific and global products: A two-stage stochastic program with additional demand after recourse, Naval Res Logistics 50 (2003), 438–461.
[5] R.K.-M. Cheung and W.B. Powell, Stochastic programs over trees with random arc capacities, Networks 24 (1994), 161–175.
[6] R.K.-M. Cheung and W.B. Powell, An algorithm for multistage dynamic networks with random arc capacities, with an application to dynamic fleet management, Oper Res 44 (1996), 951–963.
[7] M.A.H. Dempster, N. Hicks Pedron, E.A. Medova, J.E. Scott, and A. Sembos, Planning logistics operations in the oil industry, J Oper Res Soc 51 (2000), 1271–1288.
[8] D. Dentcheva, A. Prékopa, and A. Ruszczyński, Concavity and efficient points of discrete distributions in probabilistic programming, Math Prog Ser A 89 (2000), 55–77.
[9] L.F. Escudero, P.V. Kamesam, A.J. King, and R.J.-B. Wets, Production planning via scenario modelling, Ann Oper Res 43 (1993), 311–335.

[10] http://www.globalsecurity.org/military/library/policy/army/fm/3-09-21/ch.7.htm#sec3par1.

[11] A. Gupta, C.D. Maranas, and C.M. McDonald, Mid-term supply chain planning under demand uncertainty: Customer demand satisfaction and inventory management, Comput Chem Eng 24 (2000), 2613–2621.

[12] M. Kress, Operational logistics: The art and science of sustaining military operations, Kluwer Academic, Boston, 2002.

[13] http://www.pakdef.info/pakmilitary/army/artillery/m109.html.

[14] D. Pisinger and P. Toth, "Knapsack problems," Handbook of combinatorial optimization, D.Z. Du and P. Pardalos (Editors), Kluwer Academic, Norwell, MA, 1998, pp. 299–428.

[15] A. Ruszczyński, Probabilistic programming with discrete distributions and precedence constrained knapsack polyhedra, Math Prog Ser A 93 (2002), 195–215.

[16] S. Sen, Relaxations for probabilistically constrained programs with discrete random variables, Oper Res Lett 11 (1992), 81–86.